

# Package ‘recountmethylation’

October 14, 2021

**Version** 1.2.3

**Title** Access and analyze DNA methylation database compilations

**Description** Access cross-study compilations of DNA methylation array databases. Database files can be downloaded and accessed using provided functions. Background about database file types (HDF5 and HDF5-SummarizedExperiment), SummarizedExperiment classes, and examples for data handling, validation, and analyses, can be found in the package vignettes. Note the disclaimer on package load, and consult the main manuscript for further info.

**License** Artistic-2.0

**Encoding** UTF-8

**URL** <https://github.com/metamaden/recountmethylation>

**BugReports** <https://github.com/metamaden/recountmethylation/issues>

**LazyData** FALSE

**Depends** R (>= 4.0.0)

**Imports** minfi, HDF5Array, rhdf5, S4Vectors, utils, methods, RCurl,  
R.utils, BiocFileCache

**Suggests** knitr, testthat, ggplot2, gridExtra, rmarkdown, BiocStyle,  
GenomicRanges, limma, ExperimentHub, AnnotationHub

**VignetteBuilder** knitr

**biocViews** DNAMethylation, Epigenetics, Microarray, MethylationArray,  
ExperimentHub

**RoxygenNote** 7.1.1

**git\_url** <https://git.bioconductor.org/packages/recountmethylation>

**git\_branch** RELEASE\_3\_13

**git\_last\_commit** 3557cec

**git\_last\_commit\_date** 2021-10-11

**Date/Publication** 2021-10-14

**Author** Sean K Maden [cre, aut] (<<https://orcid.org/0000-0002-2212-4894>>),  
Reid F Thompson [aut] (<<https://orcid.org/0000-0003-3661-5296>>),  
Kasper D Hansen [aut] (<<https://orcid.org/0000-0003-0086-0687>>),  
Abhinav Nellore [aut] (<<https://orcid.org/0000-0001-8145-1484>>)

**Maintainer** Sean K Maden <maden@ohsu.edu>

## R topics documented:

data_mdpost . . . . .	2
gds_idat2rg . . . . .	3
gds_idatquery . . . . .	4
getdb . . . . .	5
getrg . . . . .	6
get_rmdl . . . . .	8
get_servermatrix . . . . .	9
hread . . . . .	10
matchds_1to2 . . . . .	11
rgse . . . . .	12
servermatrix . . . . .	12
smfilt . . . . .	13
<b>Index</b>	<b>15</b>

---

data_mdpost	<i>Retrieve all available sample metadata from an HDF5 database.</i>
-------------	--

---

### Description

Retrieve all available sample metadata in a dataset from an HDF5 database. Returns data in meta-data dataset "dsn" contained in an h5 file located at path "dbn."

### Usage

```
data_mdpost(dbn = "remethdb2.h5", dsn = "mdpost")
```

### Arguments

dbn	Path to h5 HDF5 database file.
dsn	Name or group path to HDF5 dataset containing the sample metadata and learned annotations.

### Value

data.frame of available sample metadata.

### See Also

hread()

**Examples**

```
path <- system.file("extdata", "h5test", package = "recountmethylation")
fn <- list.files(path)
dbpath <- file.path(path, fn)
mdp <- data_mdpost(dbn = dbpath, dsn = "mdpost")
dim(mdp) # [1] 2 19
```

gds\_idat2rg

*Get IDATs as an RGChannelSet from GEO/GDS***Description**

Queries and downloads GSM IDAT files in GEO Data Sets db, then returns the assay data as an "RGChannelSet", calling `gds_idatquery()` then `minfi::read.metharray()`.

**Usage**

```
gds_idat2rg(
  gsmvi,
  rmdl = TRUE,
  ext = "gz",
  dfp = "./idats/",
  burl = paste0("ftp://ftp.ncbi.nlm.nih.gov/", "geo/samples/"),
  silent = TRUE
)
```

**Arguments**

<code>gsmvi</code>	A vector of GSM IDs (alphanumeric character strings).
<code>rmdl</code>	Whether to remove downloaded IDAT files when finished (default TRUE).
<code>ext</code>	Extension for downloaded files (default "gz").
<code>dfp</code>	Destination for IDAT downloads.
<code>burl</code>	Base URL string for the IDAT query (default "ftp://ftp.ncbi.nlm.nih.gov/geo/samples/").
<code>silent</code>	Whether to suppress warnings on download removal (default TRUE).

**Value**

An RGChannelSet object

**See Also**

`gds_idatquery()`, `read.metharray()`

**Examples**

```
gsmvi <- c("GSM2465267", "GSM2814572")
fpath <- file.path(tempdir(), "gds_idat2rg_example")
rg <- try(gds_idat2rg(gsmvi, dfp = fpath))
```

---

`gds_idatquery`*Query and download IDATs from GEO Data Sets*

---

**Description**

Queries GEO Data Sets for IDATs, and downloads available IDATs. This uses anticipated string pattern to construct the URL path for the query. IDATs are detected from the supplement for a GSE record.

**Usage**

```
gds_idatquery(  
  gsmvi,  
  ext = "gz",  
  expand = TRUE,  
  verbose = FALSE,  
  dfp = "idats",  
  burl = paste0("ftp://ftp.ncbi.nlm.nih.gov/geo/samples/")  
)
```

**Arguments**

<code>gsmvi</code>	Vector of valid GSM IDs.
<code>ext</code>	Filename extension.
<code>expand</code>	Whether to expand compressed files.
<code>verbose</code>	Whether to show verbose messages (default FALSE).
<code>dfp</code>	Destination directory for downloads.
<code>burl</code>	Base URL string for RCurl query.

**Value**

Lists the basename paths and filenames of IDATs downloaded.

**Examples**

```
query <- try(gds_idatquery(gsmvi = c("GSM2465267", "GSM2814572")))
```

---

getdb                      *Access database files.*

---

## Description

Combines download and load functions for databases. If the "namematch" argument isn't provided, the latest available file is downloaded. All files include metadata for the available samples.

There are 6 functions. Functions with "h5se" access HDF5-SummarizedExperiment files, and "h5" functions access HDF5 databases. The 4 h5se functions are "rg" (RGChannelSet), "gm" (MethylSet), "gr" (GenomicRatioSet), and "test" (data for 2 samples from "gr"). The 2 h5 functions are "rg" (red and green signal datasets), and "test" (data for 2 samples from "rg"). See vignette for details about file types and classes.

## Usage

```
getdb_h5se_test(  
  platform = NULL,  
  dfp = NULL,  
  namematch = "remethdb-h5se-gr-test.*",  
  verbose = FALSE  
)
```

```
getdb_h5_test(  
  platform = NULL,  
  namematch = "remethdb-h5_rg-test_.*",  
  dfp = NULL,  
  verbose = FALSE  
)
```

```
getdb_h5se_gr(  
  platform = c("hm450k", "epic"),  
  dfp = NULL,  
  namematch = "remethdb_h5se-gr_.*",  
  verbose = FALSE  
)
```

```
getdb_h5se_gm(  
  platform = c("hm450k", "epic"),  
  dfp = NULL,  
  namematch = "remethdb_h5se-gm_.*",  
  verbose = FALSE  
)
```

```
getdb_h5se_rg(  
  platform = c("hm450k", "epic"),  
  dfp = NULL,
```

```

    namematch = "remethdb-h5se_rg_.*",
    verbose = FALSE
  )

  getdb_h5_rg(
    platform = c("hm450k", "epic"),
    dfp = NULL,
    namematch = "remethdb-h5_rg_.*",
    verbose = FALSE
  )

```

### Arguments

platform	Valid supported DNAm array platform type. Currently either "epic" for EPIC/HM850K, or "hm450k" for HM450K.
dfp	Folder to search for database file (optional, if NULL then searches cache dir specified by BiocFileCache).
namematch	Filename pattern to match when searching for database (see defaults).
verbose	Whether to return verbose messages (default FALSE).

### Value

Either a SummarizedExperiment object for h5se functions, or a file path for h5 functions.

### See Also

`get_rmdl()`

### Examples

```
h5 <- getdb_h5_test(dfp = tempdir())
```

---

getrg

*Query and store data from h5 file signal tables*

---

### Description

Queries signal datasets in an h5 HDF5 database file. Handles identity queries to rows (GSM IDs) or columns (bead addresses). Returns query matches either as a list of datasets or a single RGChannelSet, with option to include sample metadata.

**Usage**

```

getrg(
  dbn,
  gsmv = NULL,
  cgv = NULL,
  data.type = c("se"),
  dsv = c("redsignal", "greensignal"),
  all.gsm = FALSE,
  all.cg = TRUE,
  metadata = TRUE,
  md.dsn = "mdpost",
  verbose = FALSE
)

```

**Arguments**

dbn	Name of the HDF5 database file.
gsmv	Vector valid GSM IDs (rows) to query, either NULL or vector of length > 2 valid GSM IDs, or "all.gsm" should be TRUE.
cgv	Vector of valid bead addresses (columns) to query in the signal datasets (default NULL).
data.type	Format for returned query matches, either as datasets "df" or RGChannelSet "se" object.
dsv	Vector of raw signal datasets or group paths to query, including both the red channel 'redsignal' and green channel 'greensignal' datasets.
all.gsm	Whether to query all available GSM IDs.
all.cg	Whether to query all available CpG probe addresses.
metadata	Whether to access available postprocessed metadata for queried samples.
md.dsn	Name of metadata dataset in h5 file.
verbose	Whether to post status messages.

**Value**

Returns either an RGChannelSet or list of data.frame objects from dataset query matches.

**See Also**

rgse()

**Examples**

```

path <- system.file("extdata", "h5test", package = "recountmethylation")
fn <- list.files(path)
dbpath <- file.path(path, fn)
rg <- getrg(dbn = dbpath, all.gsm = TRUE, metadata = FALSE)
dim(rg) # [1] 11162    2

```

```
class(rg)
# [1] "RGChannelSet"
# attr(,"package")
# [1] "minfi"
```

---

get\_rmdl

*Get DNAm assay data.*


---

## Description

Uses RCurl to download the latest HDF5-SummarizedExperiment or HDF5 database compilation files objects from the server. Calls servermatrix and performs various quality checks to validate files and downloads. This function is wrapped in the getdb() set of functions (type ‘?getdb’ for details).

## Usage

```
get_rmdl(
  which.class = c("rg", "gm", "gr", "test"),
  which.type = c("h5se", "h5"),
  which.platform = c("hm450k", "epic"),
  fn = NULL,
  dfp = "downloads",
  url = "https://methylation.recount.bio/",
  show.files = FALSE,
  download = TRUE,
  sslver = FALSE,
  verbose = TRUE
)
```

## Arguments

which.class	Either "rg", "gm", "gr", or "test" for RGChannelSet, MethylSet, GenomicRatioSet, or 2-sample subset.
which.type	Either "h5se" for an HDF5-SummarizedExperiment or "h5" for an HDF5 database.
which.platform	Supported DNAm array platform type. Currently supports either "epic" for EPIC/HM850K, or "hm450k" for HM450K.
fn	Name of file on server to download (optional, default NULL).
dfp	Download destination directory (default "downloads").
url	The server URL to locate files for download.
show.files	Whether to print server file data to console (default FALSE).
download	Whether to download (TRUE) or return queried filename (FALSE).
sslver	Whether to use server certificate check (default FALSE).
verbose	Whether to return verbose messages (default TRUE).



**Value**

New filepath to dir containing the downloaded data.

**See Also**

servermatrix(), getURL(), loadHDF5SummarizedExperiment(), h5ls()

**Examples**

```
# prints file info from server:
path <- try(get_rmdl(which.class = "test", which.type = "h5se",
show.files = TRUE, download = FALSE))
```

---

get\_servermatrix      *get\_servermatrix*

---

**Description**

Get a matrix of server files. If the RCurl call fails, a matrix is loaded from the stored package files at 'sm\_path'.

**Usage**

```
get_servermatrix(
  dn = NULL,
  sslver = FALSE,
  printmatrix = TRUE,
  url = "https://methylation.recount.bio/",
  verbose = FALSE,
  sm_path = system.file("extdata", "servermatrix_rda", package = "recountmethylation")
)
```

**Arguments**

dn	Server data returned from RCurl (default NULL).
sslver	Whether to use SSL certificate authentication for server connection (default FALSE).
printmatrix	Whether to print the data matrix to console (default TRUE).
url	Server website url (default "https://methylation.recount.bio/").
verbose	Whether to show verbose messages (default FALSE).
sm_path	Path to the servermatrix_rda dir containing the stored servermatrix files (default: system.file...).

**Value**

Matrix of server files and file metadata

**See Also**

servermatrix, get\_rmdl, smfilt

**Examples**

```
sm <- get_servermatrix(url = "")
```

---

hread

*Query and store an HDF5 dataset on row and column indices.*

---

**Description**

Connect to an HDF5 database h5 file with rhdf5::h5read(). Returns the subsetted data.

**Usage**

```
hread(ri, ci, dsn = "redsignal", dbn = "remethdb2.h5")
```

**Arguments**

ri	Row indices in dataset.
ci	Column indices in dataset.
dsn	Name of dataset or group of dataset to connect with.
dbn	Path to h5 database file.

**Value**

HDF5 database connection object.

**Examples**

```
# Get tests data pointer
path <- system.file("extdata", "h5test", package = "recountmethylation")
fn <- list.files(path)
dbpath <- file.path(path, fn)
# red signal, first 2 assay addr, 3 samples
reds <- hread(1:2, 1:3, d = "redsignal", dbn = dbpath)
dim(reds) # [1] 2 3
```

---

 matchds\_1to2

---

*Match two datasets on rows and columns*


---

### Description

Match 2 datasets using the character vectors of row or column names. This is used to assemble an "RGChannelSet" from a query to an h5 dataset.

### Usage

```
matchds_1to2(
  ds1,
  ds2,
  mi1 = c("rows", "columns"),
  mi2 = c("rows", "columns"),
  subset.match = FALSE
)
```

### Arguments

ds1	First dataset to match
ds2	Second dataset to match
mi1	Match index of ds1 (either "rows" or "columns")
mi2	Match index of ds2 (either "rows" or "columns")
subset.match	If index lengths don't match, match on the common subset instead

### Value

A list of the matched datasets.

### Examples

```
# get 2 data matrices
ds1 <- matrix(seq(1, 10, 1), nrow = 5)
ds2 <- matrix(seq(11, 20, 1), nrow = 5)
rownames(ds1) <- rownames(ds2) <- paste0("row", seq(1, 5, 1))
colnames(ds1) <- colnames(ds2) <- paste0("col", c(1, 2))
ds2 <- ds2[rev(seq(1, 5, 1)), c(2, 1)]
# match row and column names
lmatched <- matchds_1to2(ds1, ds2, mi1 = "rows", mi2 = "rows")
lmatched <- matchds_1to2(lmatched[[1]], lmatched[[2]], mi1 = "columns",
  mi2 <- "columns")
# check matches
ds1m <- lmatched[[1]]
ds2m <- lmatched[[2]]
identical(rownames(ds1m), rownames(ds2m))
identical(colnames(ds1m), colnames(ds2m))
```

---

rgse	<i>Form an RGChannelSet from a list containing signal data matrices</i>
------	---

---

**Description**

Forms an RGChannelSet from signal data list. This is called by certain queries to h5 files.

**Usage**

```
rgse(ldat, verbose = FALSE)
```

**Arguments**

ldat	List of raw signal data query results. Must include 2 data.frame objects named "redsignal" and "greensignal."
verbose	Whether to post status messages.

**Value**

Returns a RGChannelSet object from raw signal dataset queries.

**See Also**

getrg(), RGChannelSet()

**Examples**

```
path <- system.file("extdata", "h5test", package = "recountmethylation")
fn <- list.files(path)
dbpath <- file.path(path, fn)
rg <- getrg(dbn = dbpath, all.gsm = TRUE, metadata = FALSE)
dim(rg) # [1] 11162    2
class(rg)
# [1] "RGChannelSet"
# attr(,"package")
# [1] "minfi"
```

---

servermatrix	<i>servermatrix</i>
--------------	---------------------

---

**Description**

Called by get\_rmdl() to get a matrix of database files and file info from the server. Verifies valid versions and timestamps in filenames, and that h5se directories contain both an assays and an se.rds file.

**Usage**

```
servermatrix(
  dn = NULL,
  sslver = FALSE,
  printmatrix = TRUE,
  url = "https://methylation.recount.bio/",
  verbose = FALSE
)
```

**Arguments**

dn	Server data returned from RCurl (default NULL).
sslver	Whether to use SSL certificate authentication for server connection (default FALSE).
printmatrix	Whether to print the data matrix to console (default TRUE).
url	Server website url (default "https://methylation.recount.bio/").
verbose	Whether to show verbose messages (default FALSE).

**Value**

Matrix of server files and file metadata

**See Also**

get\_rmdl, smfilt

**Examples**

```
dn <- "remethdb-h5se_gr-test_0-0-1_1590090412 29-May-2020 07:28 -"
sm <- get_servermatrix(dn = dn)
```

---

smfilt

*smfilt*

---

**Description**

Filters the data matrix returned from servermatrix().

**Usage**

```
smfilt(sm, typesdf = NULL)
```

**Arguments**

sm	Data matrix returned from servermatrix().
typesdf	Data.frame containing database file info for dm filters.

**Value**

Filtered data matrix of server file info.

**See Also**

`get_rmdl`, `servermatrix`

**Examples**

```
dm <- matrix(c("remethdb_h5-rg_epic_0-0-2_1589820348.h5", "08-Jan-2021",  
"09:46", "66751358297"), nrow = 1)  
smfilt(dm)
```

# Index

[data\\_mdpost](#), [2](#)

[gds\\_idat2rg](#), [3](#)

[gds\\_idatquery](#), [4](#)

[get\\_rmdl](#), [8](#)

[get\\_servermatrix](#), [9](#)

[getdb](#), [5](#)

[getdb\\_h5\\_rg \(getdb\)](#), [5](#)

[getdb\\_h5\\_test \(getdb\)](#), [5](#)

[getdb\\_h5se\\_gm \(getdb\)](#), [5](#)

[getdb\\_h5se\\_gr \(getdb\)](#), [5](#)

[getdb\\_h5se\\_rg \(getdb\)](#), [5](#)

[getdb\\_h5se\\_test \(getdb\)](#), [5](#)

[getrg](#), [6](#)

[hread](#), [10](#)

[matchds\\_1to2](#), [11](#)

[rgse](#), [12](#)

[servermatrix](#), [12](#)

[smfilt](#), [13](#)